

# Boosting Minimalist Classifiers for Blemish Detection in Potatoes

Michael Barnes, Tom Duckett and Grzegorz Cielniak

Lincoln University, Lincoln, UK, LN6 7TS.

{mbarnes, tduckett, gcielniak}@lincoln.ac.uk

**Abstract**—This paper introduces novel methods for detecting blemishes in potatoes using machine vision. After segmentation of the potato from the background, a pixel-wise classifier is trained to detect blemishes using features extracted from the image. A very large set of candidate features, based on statistical information relating to the colour and texture of the region surrounding a given pixel, is first extracted. Then an adaptive boosting algorithm (AdaBoost) is used to automatically select the best features for discriminating between blemishes and non-blemishes. With this approach, different features can be selected for different potato varieties, while also handling the natural variation in fresh produce due to different seasons, lighting conditions, etc. The results show that the method is able to build “minimalist” classifiers that optimise detection performance at low computational cost. In experiments, minimalist blemish detectors were trained for both white and red potato varieties, achieving 89.6% and 89.5% accuracy respectively.

## I. INTRODUCTION

Potatoes (*Solanum* sp. commonly *S. tuberosum*), with an estimated worldwide production of over 300,000,000 tonnes in 2005 [1], account for 70-80% of the carbohydrate consumed in the UK with millions of tons harvested worldwide. For the fresh market the main factor affecting consumer preference is physical appearance and, to maximise return, great effort is expended ensuring that the appearance best matches a particular market. Most potatoes are still sorted by hand. A group of human potato graders will each process around 2 tonnes of potatoes per hour [2], so on average each human decides whether to reject one medium baking potato every 0.45 seconds. Problems with manual sorting include the subjectivity, fatigue and high cost of human inspectors. Therefore there is considerable motivation for a machine vision application with both speed and accuracy considerations.

### A. Potato blemishes

There are a number of diseases affecting potato tubers that, although superficial and generally of little or no health consequence to humans, strongly and negatively influence consumer choice. These include black dot, silver scurf, powdery scab, common scab, and skin spot. The fungal species of *Rhizoctonia Solani* also causes significant skin blemish as black scurf and elephant hide. Other forms of blemish include physical damage, e.g. growth cracks, mechanical damage and slug damage as well as physiological blemishes, e.g. greening and sprouting.

These conditions present a variety of different coloured, sized and textured symptoms on the skin surface. Such diverse

visual information provides us with a rich source of indicators that can be used for training an automatic blemish detector.

### B. Related work

In typical machine vision systems for quality analysis of food products, there are several major steps: after pre-processing (e.g. to segment the object of interest from the background), image features are extracted that summarise important qualities of the object, then a pattern recognition system is used to categorise the input data. For example, [3] introduced a system for sorting sweet tamarind, by measuring the size and shape of tamarind pods as well as detecting defects in the form of broken pods. Thresholded intensity values were used to distinguish blemishes from non-blemishes. [4] introduced visual inspection methods for pasteurised cheese. They also used thresholding to detect ingredients such as chives, and developed methods to measure the distribution and quantity of the detected ingredients. [5] developed methods to distinguish between blemishes in apples and healthy apples with visible stem or calyx. Images were recorded using special filters to restrict the observed light frequencies, then various features including statistical moments and shape features were used for pattern recognition. [6] developed the VeggieVision system, using HSV-colour and texture histograms to classify different types of fruit and vegetables, with application to a supermarket check-out for automatic produce recognition. [7] developed a machine vision system for automatic descriptive sensory evaluation of meals, where a neural network was trained to mimic the opinion of human experts in describing the sensory attributes of a prototypical meal.

A limitation of typical systems is that the set of image features for pattern recognition has to be designed by the system engineer to work with a specific configuration of produce, imaging system and operating conditions. Such systems typically do not generalise well to other configurations, where the required image features may well differ from those used to design the original system. The novelty of the approach presented in this paper involves the use of an adaptive boosting algorithm (AdaBoost [8]) to automatically select good features for a particular pattern recognition task. A minimal set of features is selected from a very large set of *candidate features*, which measure statistical properties of the colour and texture distribution of the image region surrounding a given pixel. Thus the *selected features* used to build the final pattern recognition system are optimised for a particular application

by learning from examples, and the system can be retrained to select a different set of features in order to accommodate different varieties of produce, seasonal variations, etc.

The objective of this paper is to introduce an automatic method for detecting blemishes in digital images of potatoes. The system developed should be trainable, so that it can work with different varieties of potatoes and variations in seasons, lighting conditions, etc. A human expert is required to mark up areas of blemishes and non-blemishes in a set of training images. After training, the system should be able to classify individual pixels as blemishes or non-blemishes with high accuracy. A further objective, with eventual deployment in industrial settings in mind, is to enable real-time processing of images (possibly in rapid succession) by building “minimalist” classifiers that extract a minimal subset of all features that optimise detection performance at the lowest possible computational cost. Finally the feature selection mechanism developed should be perspicuous to human users, allowing operators to understand which features are important to distinguish blemishes from non-blemishes for different potato varieties. The contribution of this paper towards meeting these objectives is demonstrated in experiments by learning minimalist blemish detectors for both white and red potatoes, achieving 89.6% and 89.5% accuracy respectively.

## II. MATERIALS AND METHODS

### A. Image acquisition

The experimental data for this system, consisting of images of potatoes, was acquired using a colour camera (Sony DSLR-A350K) fixed above the tubers which in turn were placed on a white board. The camera was set at a distance of 60cm from the camera objective to the base on which the subjects were placed, with a focal length of 70mm and an aperture setting of F22. The resolution of the images was  $1536 \times 1024$  pixels. To reduce the effects of shadows and changing light conditions the potatoes were placed inside a white cylinder with daylight bulbs placed around the top. The equipment used to capture these images is shown in Fig. 1.

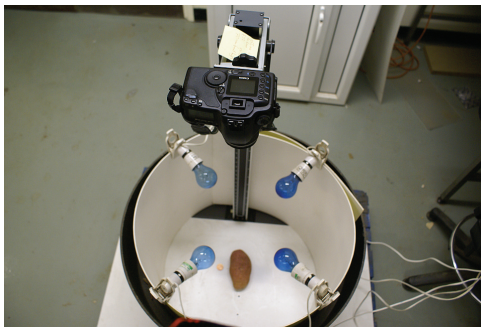


Fig. 1. The camera setup for photographing the training data at a constant distance with all-around lighting.

### B. Ground truth

There were two sets of data collected for white and red potatoes respectively, including potatoes affected by different

blemishes. The white potato data set consisted of 102 images including 19 images containing a single blemish type, 39 images with two distinct blemish types, 38 images with three blemish types and 6 images containing more than three blemish types. The most common blemishes were black dot and silver scurf, appearing in 69 and 53 images respectively, while the rarest were powdery scab, elephant hide and growth cracks, with no more than 3 images of each. The red potato data set consisted of 22 images with the most common blemishes again being black dot and silver scurf appearing in 13 and 6 images respectively as well as common scab, appearing in 12 cases. 10 red potato images had 2 different blemish types, 3 images had 3 types, 1 image had 4 types, and the remaining 8 images had only one blemish type.

To train the classifiers and test their performance, the images need to be marked up by hand to provide the “ground truth” information indicating the correct class of each pixel. The mark up process begins with a semi-automatic method for background removal, using the Magic Wand tool in Photoshop to label the image region surrounding the potato. The potato area is then hand labelled by an industry expert into regions corresponding to blemish and non-blemish. It is not necessary to label all pixels in an image: some areas of high uncertainty or ambiguity are left unmarked, and these pixels are ignored during training of the classifier. Background pixels are also omitted from the subsequent calculations.

### C. Feature extraction

The first step of the procedure is to extract different image features that should indicate the presence or absence of blemishes in a potato image. The features include statistical summaries of the whole potato and local regions centred on each pixel as well as the data of the pixel itself. The statistics used were the mean, variance and skew. Other systems have used only the mean of the region such as [9] or histograms as in [6]. The proposed system uses the RGB colour space - the original colour format of the camera output. An alternative solution would be to use the HSV colour space but this would create an additional processing overhead including colour conversion and calculation of circular statistics. Other systems use more complex hardware set-ups such as customised lighting, as is an option for the Maf-Roda Agrobotic [10] or using specific colour filters like in [5]. The code for the system software was implemented in MATLAB.

The image regions used in our experiments were squares of size  $33 \times 33$ ,  $65 \times 65$ ,  $97 \times 97$ ,  $129 \times 129$  and  $161 \times 161$ , plus the whole potato, giving 6 regions in total.

Our system uses seven colour channels; raw RGB, normalised RGB and the intensity channel. From these channels we consider the following image properties:

a) *Colour*: Intensity is especially of relevance for dark blemishes, e.g. black scurf or skin spot, while the most obvious blemish to be detected by other colour channels would be greening. The three statistical moments collected from the seven colour channels represent the first 21 features for each

region. Seven additional features describe the colour properties of the pixel itself.

b) *Edges*: An edge detector determines the rate of change of pixel values in a given neighbourhood in a specific direction. Some blemishes tend to coincide with high rates of change, such as powdery scab when the skin splits. The Sobel edge detector was used in this case with a standard  $3 \times 3$  kernel size. The edge detector was run on the same seven colour channels listed above. These statistics provide 21 features for each region and seven for the pixel itself.

c) *Range*: The range filter determines the maximum difference between pixel values in a given neighbourhood indicating the roughness of the texture. Higher values tend to correspond to rougher, potentially damaged areas of the image. The range filter was run on the same seven channels with a  $5 \times 5$  neighbourhood. The three statistical moments collected from the resulting range information provide another 21 features for each region. Seven additional features describe the range properties of the pixel itself.

In summary there are 7 colour channels  $\times$  3 feature types  $\times$  3 statistical moments making 63 features for each region and  $7 \times 3 = 21$  features for the pixel itself. All these features are used as the candidate feature set. Since there are 6 different regions, this gives  $63 \times 6 = 378$  features which, with additional 21 features for the pixel itself, gives us  $F_c = 399$  candidate features in total. These features were used as the training input to our classifier.

#### D. AdaBoost

The AdaBoost algorithm [8] is used to build a classifier, which combines results from so-called “weak” classifiers (each constructed using one of the candidate features) into one “strong” classifier that performs better than any of the weak classifiers alone. It has been used previously in the classification of apples to avoid falsely classifying apple features as blemishes [5]. The high performance of the final strong classifier is due to the emphasis put on the training examples which are most difficult to classify during the learning process. This method is called boosting. During training AdaBoost makes a number of passes, called rounds or iterations, through the training data. Each time it finds the next best feature to improve the number of correctly classified examples, prioritising those examples which were misclassified previously. In each pass one feature is selected and assigned a weight and a threshold to create a new weak classifier. The weak classifiers are then combined into a strong classifier wherein each weak classifier is given a weighted vote in the classification of a given example.

Real AdaBoost [11] is a generalisation of this algorithm that provides a lower error rate by allowing weak classifiers to vote by their individual degree of certainty instead of simply voting “yes” or “no”. It is the version used in our experiments, hereafter referred to simply as AdaBoost.

AdaBoost has been implemented on a range of tasks, including face detection [12] but also in food quality control systems, such as [5].

1) *Minimalist AdaBoost*: The AdaBoost classifier selects a set of the most useful features from all candidate features. If the training data is not normally distributed, AdaBoost will often choose the same feature for more than one weak classifier. Therefore it is of interest to see how much the classification success rate would be affected by the original candidate feature set being restricted to a subset of features, selected by AdaBoost itself. By doing so the total number of unique features required to be extracted for classification will be reduced and therefore less computational time will be required by the feature extraction stage. We refer to this subset of features as “selected features”.

Our minimalist classifier will consist of two stages both incorporating the AdaBoost algorithm: the first stage selects a feature set that will be used to train an AdaBoost classifier in the second stage. Algorithm 1 presents the Real AdaBoost algorithm as described in [13], applied with our addition of step 4, in order to limit the number of unique features used in the final classifier to a smaller number than the total number of weak classifiers allowed.

Using MATLAB we have extended the AdaBoost implementation within the GML AdaBoost Toolbox [14], to build the minimalist classifier.

### III. RESULTS AND DISCUSSION

#### A. Training and testing

When training, the minimalist classifier first chooses a number of “selected features” ( $F_s$ ). To investigate the impact of this parameter on the classification rate we used  $F_s = 1, 2, 5$  and 10. For comparison we also used  $F_s = F_c$ , the equivalent of a non-minimalist system. The number of AdaBoost rounds in the second stage was set to  $T = 40$ .

Tests were carried out using the training data on a hold-one-out basis whereby one of the 102 images is removed from the training data and used as test data instead. The testing was carried out for every image in this manner with each being tested on a classifier trained on the other 101. On average, the training data for each such experiment consisted of 8792 pixels of good potato and 8461 pixels of blemish. The success rates of the minimalist classifier for different potato colours and different values of  $F_s$  are presented in Table I. In addition, the performance of the classifier using ROC curves is presented in Fig. 2 for white potatoes and in Fig. 3 for red potatoes respectively. To determine the importance of different feature categories (i.e. colour, edge and range), the tests were carried out for different subsets of these categories. The results are presented as ROC curves in Fig. 4 for white potatoes and in Fig. 5 for red potatoes respectively.

Fig. 6 shows the output of the classifier compared to the ground truth information. Some of the disparity between the classifier output and ground truth is due to human inaccuracy at the markup stage which can be seen more clearly in Fig. 7.

#### B. Success rates

Results presented in Table I for different numbers of selected features indicate that using  $F_s = 10$  features does not

**Algorithm 1** Our implementation of the Real AdaBoost learning algorithm. [8], [11]

Given a dataset  $S = \{(x_1, y_1), \dots, (x_m, y_m)\}$  where  $x_i \in X$  and  $y_i \in \{-1, +1\}$ , the weak classifier pool  $K$ , containing all possible weak classifiers from  $F_c$  candidate features, a specific number of weak classifiers to be chosen  $T$  and a maximum number of unique features to be used to choose these weak classifiers  $F_s$ . Initialise the sample distribution  $D_1(i) = 1/m$

For  $t = 1, \dots, T$

1. For each weak classifier  $h$  in  $K$  do:
  - a. Partition  $X$  into several disjoint blocks  $X_1, \dots, X_n$
  - b. Using the weights in distribution  $D_t$  calculate

$$W_t^j = P(x_i \in X_j, y_i = l) = \sum_{i: x_i \in X_j, y_i = l} D_t(i)$$

Where  $l = \pm 1$

- c. Set the output of  $h$  on each  $X_j$  as

$$\forall x \in X_j, h(x) = \frac{1}{2} \ln \left( \frac{W_{+1}^j + \epsilon}{W_{-1}^j + \epsilon} \right)$$

- d. Calculate the normalisation factor

$$Z = 2 \sum_j \sqrt{W_{+1}^j W_{-1}^j}$$

2. Select the  $h_t$  Minimising  $Z$  i.e.

$$Z_t = \min_{h \in K} Z$$

$$h_t = \arg \min_{h \in K} Z$$

3. Update the sample distribution

$$D_{t+1}(i) = D_t(i) \exp[-y_i h_t(x_i)]$$

and normalise  $D_{t+1}$  to give a probability distribution function.

4. Count the number of unique features used by all weak classifiers. If the total equals  $F_s$  then update  $K$  to only contain weak classifiers which use already selected features.

The final strong classifier  $H$  is

$$H(x) = \text{sign} \left[ \sum_{t=1}^T h_t(x) - b \right]$$

The confidence of  $H$  is defined as

$$\text{Conf}_H(x) = \left[ \sum_{t=1}^T h_t(x) - b \right]$$

negatively impact the performance of the classifier resulting in success rates of 89.6% and 89.5% for white and red potatoes respectively. In our tests the minimalist classifier provided comparable results to the non-minimalist classifier (i.e.  $F_s = F_c$ ) for both white and red potatoes, even slightly outperforming it. This may be due to overfitting [15]. Further reducing  $F_s$  to 5 still gives satisfactory results but the performance drops noticeably below that number. The difference in performance between classifiers using different values of  $F_s$  can also be seen clearly in the ROC curves (Figs. 2 and 3).

The results of using different subsets of feature categories are shown in Table II. Using only colour gives a result of 86.3% accuracy for white potatoes and 81.3% for red potatoes. Adding edge features gives an increase of classification rate up to 88.4% and 86.8% for white and red potatoes respectively. On the other hand range features give an increase of

classification rate up to 90.1% and 86.4% for white and red potatoes respectively. This indicates that range features provide more relevant information than edges. Including all features does not greatly affect the classification rates resulting in 89.6% and 89.5% for white and red potatoes respectively. The difference in performance between classifiers using different feature categories can also be seen clearly in the ROC curves (Figs. 4 and 5).

Features	Success Rate	
	White Potato	Red Potato
$F_s = 1$	82.7%	84.8%
$F_s = 2$	87.6%	84.5%
$F_s = 5$	89.8%	88.7%
$F_s = 10$	89.6%	89.5%
$F_s = F_c$	89.7%	88.7%

TABLE I  
SUCCESS RATES FOR DIFFERENT NUMBERS OF SELECTED FEATURES  $F_s$ .

Features	Success Rate	
	White Potato	Red Potato
colour only	86.3%	81.3%
colour and edges	88.4%	86.8%
colour and range	90.1%	86.4%
colour, range and edges	89.6%	89.5%

TABLE II  
SUCCESS RATES FOR DIFFERENT SUBSETS OF FEATURE CATEGORIES ( $F_s = 10$ ).

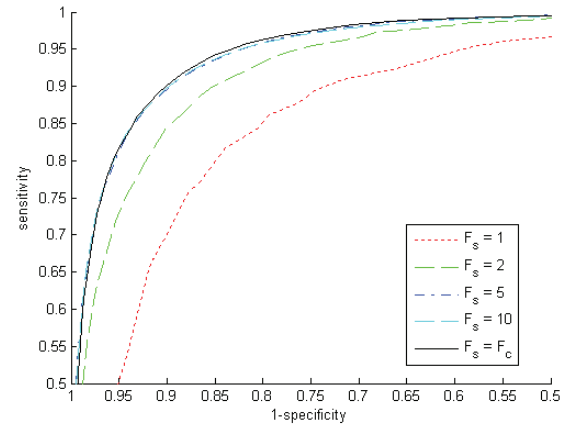


Fig. 2. ROC curves for different numbers of selected features  $F_s$  tested on white potatoes.

### C. Preferred features

The minimalist classifier selects different features for red potatoes to those selected for whites, as seen in Tables III and IV. For white potatoes the classifier selects more features pertaining to the red colour channels. Both classifiers select more texture than colour features and select the output of the edge detector on the raw red colour channel as the first feature.



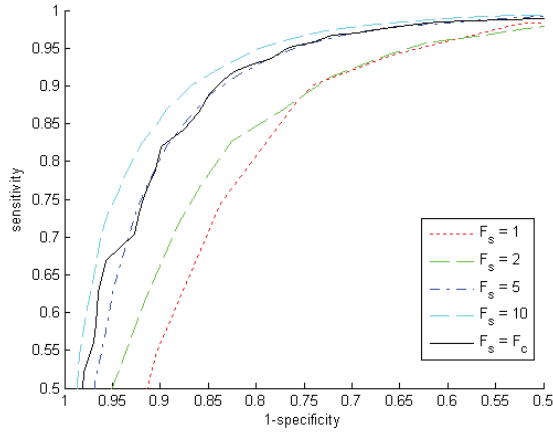


Fig. 3. ROC curves for different numbers of selected features  $F_s$  tested on red potatoes.

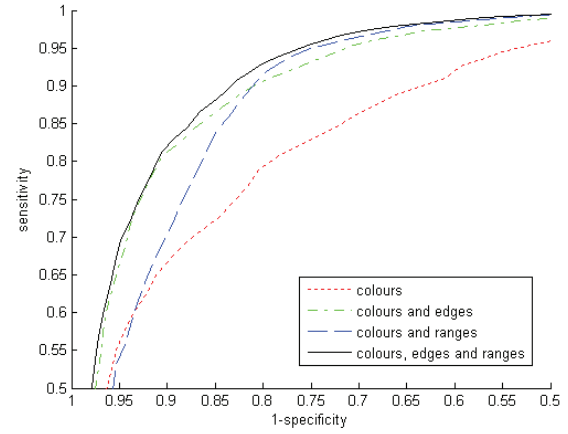


Fig. 5. ROC curves for different subsets of feature categories tested on red potatoes.

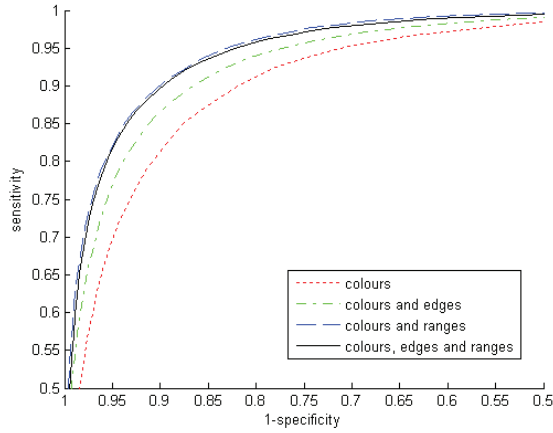


Fig. 4. ROC curves for different subsets of feature categories tested on white potatoes.

#### D. Classifier output

Figs. 6 and 7 give good examples of the output of the classifier when run on an individual potato image. A number of disagreements between ground truth and classification results were located on the edges of ground-truthed blemishes, visible

Rank	Region	Feature Type	Statistical Moment
1	$33 \times 33$	edge red	var
2	pixel	red	-
3	$33 \times 33$	range normalised red	skew
4	pixel	range red	-
5	$161 \times 161$	range red	skew
6	$33 \times 33$	range blue	skew
7	$97 \times 97$	range red	mean
8	$33 \times 33$	range green	skew
9	whole	edge normalised blue	var
10	whole	edge normalised red	var

TABLE III

THE FIRST TEN SELECTED FEATURES FOR A MINIMALIST CLASSIFIER USING WHITE POTATOES.

Rank	Region	Feature Type	Statistical Moment
1	$65 \times 65$	edge red	var
2	$65 \times 65$	normalised blue	mean
3	$33 \times 33$	range red	mean
4	$33 \times 33$	range intensity	skew
5	whole	edge normalised green	var
6	pixel	red	-
7	$161 \times 161$	edge normalised green	var
8	whole	normalised red	var
9	pixel	range red	-
10	$129 \times 129$	range blue	skew

TABLE IV

THE FIRST TEN SELECTED FEATURES FOR A MINIMALIST CLASSIFIER USING RED POTATOES.

in Fig. 6. Many of these may be due to human error in the markup stage, which can be seen as symptomatic of the problem which this research sets out to solve, that of human assessment of blemishes being subjective and prone to error. A machine vision system is likely to be more accurate than the human who produces the ground truth. This is especially noticeable in Fig. 7 where an area has been marked in the ground truth as being affected by black dot. The errors are due to the ground truth covering the whole area which is speckled with black dot blemish, while the classifier is able to detect the blemish pixel by pixel. Therefore a portion of the reported error rates is caused by inaccurate ground-truthing rather than misclassification.

#### IV. CONCLUSIONS AND FURTHER WORK

The presented results show that an AdaBoost based system is able to build minimalist classifiers that optimise detection performance at low computational cost. A minimalist classifier using only ten selected features achieves success rates of 89.6% for whites and 89.5% for red potatoes. The use of AdaBoost to build minimalist classifiers provides comparable and sometimes slightly better results than simply providing the whole feature set. It is possible that this latter fact is related to AdaBoost's vulnerability to overfitting [15].

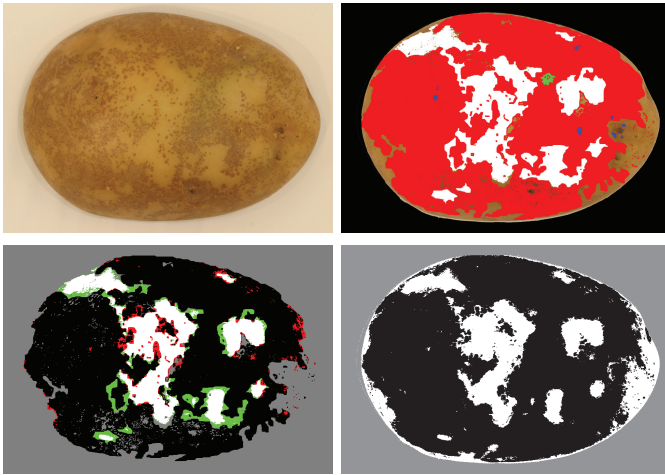


Fig. 6. Example images (clockwise from top left): first an original photograph, then a ground truthed image, then a binary image showing the classifier's view of the potato, showing pixels classified as blemish in black and good potato in white. Finally an error image, showing false positive results in red and false negative results in green.

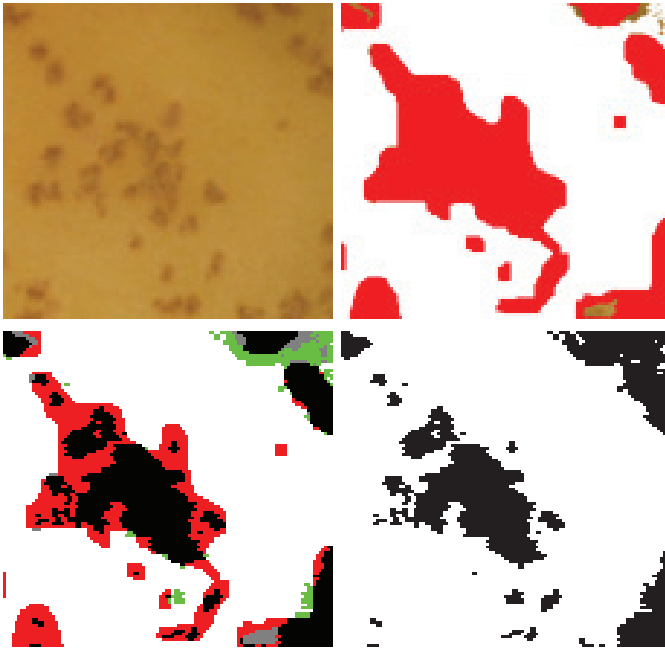


Fig. 7. Zoomed-in view of the middle of the images in Fig. 6 showing that some errors are being caused by imprecise ground truth markup.

Some initial research has suggested it might be possible to replace the fixed square regions used in our experiments with regions from segmentation algorithms (e.g. normalised cuts [16]). Also the use of textons [17] as additional texture information might improve results. However the processing time needed to follow the method used in [17] made it unappealing to pursue for this time-critical application. Further tests have yet to confirm a possible gain in accuracy or speed of these approaches.

The next stage in this research involves the identification of specific blemishes rather than just a binary classification

of blemish versus non-blemish, and investigating solutions to human inaccuracy in ground-truthing.

#### ACKNOWLEDGMENT

This work was partly funded by the Potato Council Ltd who also provided expert assistance in the field of potato science. Thanks especially to Graeme Stroud, Glyn Harper and Adrian Cunningham.

#### REFERENCES

- [1] Food and Agriculture Organisation, "FAO Statistics," Online, Food and Agriculture Organisation of the United Nations, 2005, (<http://faostat.fao.org>).
- [2] R. J. Herbert Ltd, "Meeting with representatives of R. J. Herbert Ltd," October 2008, (<http://www.rjherbert.co.uk/>).
- [3] B. Jarimopas and N. Jaisin, "An experimental machine vision system for sorting sweet taramind," *Journal of Food Engineering*, vol. 89, pp. 291 – 297, 2008.
- [4] T. Jelinski, C. jin Du, D.-W. Sun, and J. Fornal, "Inspection of the distribution and amount of ingredients in pasteurized cheese by computer vision," *Journal of Food Engineering*, vol. 83, pp. 3 – 9, 2007.
- [5] D. Unay and B. Gosselin, "Stem and calyx recognition on 'jonagold' apples by pattern recognition," *Journal of Food Engineering*, vol. 78, pp. 597 – 605, 2006.
- [6] R. Bolle, J. Connell, N. Haas, R. Mohan, and G. Taubin, "Veggievision: A produce recognition system," in *Proceedings of the 3rd IEEE Workshop on Applications of Computer Vision (WACV '96)*. Washington, DC, USA: IEEE Computer Society, 1996, p. 244.
- [7] P. Munkevik, G. Hall, and T. Duckett, "A computer vision system for appearance-based descriptive sensory evaluation of meals," *Journal of Food Engineering*, vol. 78, pp. 246 – 256, 2007.
- [8] Y. Freund and R. Schapire, "A short introduction to boosting," *Journal of Japanese Society for Artificial Intelligence*, vol. 14 (5), pp. 771 – 780, September 1999.
- [9] D. Tsai and Y. Tsai, "Rotation-invariant pattern matching with colouring projection," *Pattern Recognition*, vol. 35, pp. 131–141, 2002.
- [10] Maf Roda Group, "Meeting with Maf-Roda personnel," October 2008, (<http://www.maf-roda.com/>).
- [11] R. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," in *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, 1998, pp. 80–91.
- [12] B. Wu, H. Ai, C. Huang, and S. Lao, "Fast rotation invariant multi-view face detection based on real adaboost," in *In Sixth IEEE International Conference on Automatic Face and Gesture Recognition*, 2004, pp. 79–84.
- [13] C. Huang, B. Wu, H. Ai, and S. Lao, "Omni-directional face detection based on Real AdaBoost," in *International Conference of Computer Vision*, 2005.
- [14] A. Vezhnevets. GML AdaBoost MATLAB Toolbox. Graphics and Media Laboratory. Computer Science Department, Moscow State University, Moscow, Russian Federation. [Online]. Available: <http://research.graphicon.ru>
- [15] G. Ratsch, T. Onoda, and K. R. Muller, "An improvement of AdaBoost to avoid overfitting," in *Proceedings of the International Conference on Neural Information Processing*, 1998, pp. 506–509.
- [16] G. Mori, "Guiding model search using segmentation," in *ICCV '05: Proceedings of the Tenth IEEE International Conference on Computer Vision*, vol. 2, 2005, pp. 1417 – 1423.
- [17] M. Varma and A. Zisserman, "A statistical approach to texture classification from single images," *International Journal of Computer Vision*, vol. 62, pp. 61–81, 2005.